

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problems Mailbox.**



PATENT ABSTRACTS OF JAPAN

(11) Publication number: **2000134231 A**(43) Date of publication of application: **12 . 05 . 00**

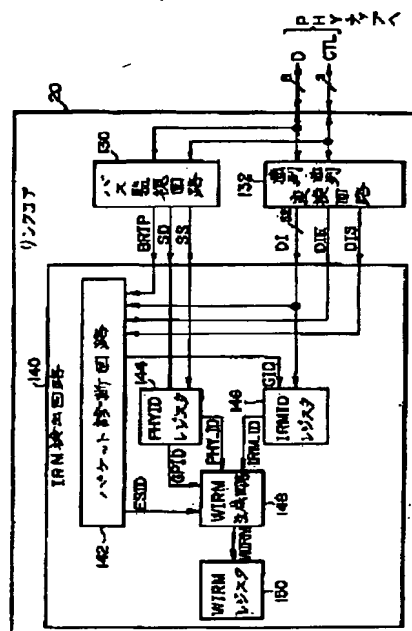
(51) Int. Cl

H04L 12/40(21) Application number: **10321539**(22) Date of filing: **27 . 10 . 98**(71) Applicant: **SEIKO EPSON CORP**(72) Inventor: **TANAKA TOSHITAKA
OGAWA TAKAHISA****(54) DATA TRANSFER CONTROLLER AND
ELECTRONIC EQUIPMENT****(57) Abstract:**

PROBLEM TO BE SOLVED: To provide a data transfer controller and electronic equipment capable of reducing the overheads of processings and transferring data at a high speed with the hardware of a small scale.

SOLUTION: A packet diagnostic circuit 142 diagnoses a self ID packet sent from respective nodes and detects the node to be the isochronous resource manager (IRM) of IEEE 1394. Then, the ID of the node is held in an IRMID register 146. A firmware recognizes the ID of the IRM by reading it. The ID of the node for which the C and L bit of the self ID packet becomes 1 is subscribed in the IRMID register 146. Whether or not it is during a self ID period is considered as the self ID packet. In the case that the IDs of the IRM and its own node match, information for indicating that its own node is the IRM is held in a WIRM register 150. By using both of GPID and ESID, the matching of the IDs of the IRM and its own node is judged.

COPYRIGHT: (C)2000,JPO



【特許請求の範囲】

【請求項1】 バスに接続される複数のノード間でのデータ転送のためのデータ転送制御装置であって、各ノードの基本情報が格納される自己識別パケットが各ノードから転送されてきた場合に、該自己識別パケットを診断し、所与の管理機能を有するマネージャになるべきノードを診断結果に基づき自動検出する回路と、検出されたノードの識別番号を該ノードの自己識別パケットの基本情報の中から取得し、該識別番号を保持する第1の保持手段とを含むことを特徴とするデータ転送制御装置。

【請求項2】 請求項1において、ノードの接続トポロジにおいてルートノードから遠いノードから順に自己識別パケットが転送されてくる場合において、自己識別パケットを転送してきたノードが、前記所与の管理機能を有するマネージャになれる資格を有するノードであると自己識別パケットの基本情報に基づき判断された場合には、該ノードの識別番号を前記第1の保持手段に上書きすることを特徴とするデータ転送制御装置。

【請求項3】 請求項1又は2において、下層から送られてくるステータス情報に基づいて自己識別期間中か否かを検出する回路を含み、自己識別期間中に送られてきたパケットを自己識別パケットと見なし、該自己識別パケットに基づいてマネージャになるべきノードの自動検出を行うことを特徴とするデータ転送制御装置。

【請求項4】 請求項1乃至3のいずれかにおいて、マネージャとなるノードの識別番号が自身のノードの識別番号と一致した場合に、自身のノードが前記所与の管理機能を有するマネージャであることを示す情報を保持する第2の保持手段を含むことを特徴とするデータ転送制御装置。

【請求項5】 請求項4において、自身のノードの識別番号が確定したことを条件にアクティブになる第1の信号と自己識別期間が終了したことを条件にアクティブになる第2の信号の両方を用いて、マネージャとなるノードの識別番号が自身のノードの識別番号と一致したか否かを判断することを特徴とするデータ転送制御装置。

【請求項6】 請求項4において、サブアクションギャップを検出したことを条件にアクティブになる第3の信号を用いて、マネージャとなるノードの識別番号が自身のノードの識別番号と一致したか否かを判断することを特徴とするデータ転送制御装置。

【請求項7】 請求項1乃至6のいずれかにおいて、前記第1の保持手段から前記所与の管理機能を有するマネージャになるノードの識別番号を読み出し、読み出された識別番号を利用してデータ転送を制御する手段を含むことを特徴とするデータ転送制御装置。

【請求項8】 請求項1乃至7のいずれかにおいて、前記所与の管理機能を有するマネージャが、IEEE1394規格におけるアイソクロナスリソースマネージャであることを特徴とするデータ転送制御装置。

【請求項9】 請求項1乃至8のいずれかのデータ転送制御装置と、前記データ転送制御装置及びバスを介して他のノードから受信したデータに所与の処理を施す装置と、処理が施されたデータを出力又は記憶するための装置とを含むことを特徴とする電子機器。

【請求項10】 請求項1乃至8のいずれかのデータ転送制御装置と、前記データ転送制御装置及びバスを介して他のノードに送信するデータに所与の処理を施す装置と、処理が施されるデータを取り込むための装置とを含むことを特徴とする電子機器。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、データ転送制御装置及びこれを含む電子機器に関する。

【0002】

【背景技術及び発明が解決しようとする課題】近年、IEEE1394と呼ばれるインターフェース規格が脚光を浴びている。このIEEE1394は、次世代のマルチメディアにも対応可能な高速シリアルバスインターフェースを規格化したものである。このIEEE1394によれば、動画像などのリアルタイム性が要求されるデータも扱うことができる。また、IEEE1394のバスには、プリンタ、スキャナ、CD-Rドライブ、ハードディスクドライブなどのコンピュータの周辺機器のみならず、ビデオカメラ、VTR、TVなどの家庭用電化製品も接続できる。このため、電子機器のデジタル化を飛躍的に促進できるものとして期待されている。

【0003】このようなIEEE1394の概要については、例えば「IEEE1394ハイ・パフォーマンス・シリアルバスの概要」(Interface Apr. 1996の1~10頁)、「PC周辺機器用バス規格群総覧」(Interface Jan. 1997の106頁~116頁)、「IEEE1394-1995 (FireWire) のリアルタイム転送モードとマルチメディア対応プロトコル」(Interface Jan. 1997の136~146頁)に開示されている。また、IEEE1394に準拠したデータ転送制御装置としては、テキサス・インスツルメンツ社製のTSB12LV31などが知られている。

【0004】しかしながら、このIEEE1394に準拠したデータ転送制御装置には次のような課題があることが判明した。

【0005】即ち、現在のIEEE1394規格によれば最大で400Mbpsの転送速度が実現可能となつて

いる。しかし、現実には、処理のオーバーヘッドの存在に起因して、システム全体の実転送速度はこれよりもかなり低くなっている。つまり、CPU上で動作するファームウェアやアプリケーションソフトが、送信データを準備したり、受信データを取り込んだりするなどの処理に多くの時間を要してしまい、バス上での転送速度が速くても、結局、高速なデータ転送を実現できない。

【0006】特に、周辺機器に組み込まれるCPUは、パーソナルコンピュータなどのホストシステムに組み込まれるCPUに比べて処理能力が低い。このため、ファームウェアやアプリケーションソフトの処理のオーバーヘッドの問題は、非常に深刻なものとなる。従って、このようなオーバーヘッドの問題を効果的に解消できる技術が望まれている。

【0007】本発明は、以上のような技術的課題に鑑みてなされたものであり、その目的とするところは、ファームウェアやアプリケーションソフトなどの処理のオーバーヘッドを軽減し、小規模なハードウェアで高速なデータ転送を実現できるデータ転送制御装置及びこれを用いられる電子機器を提供することにある。

【0008】

【課題を解決するための手段】上記課題を解決するために本発明は、バスに接続される複数のノード間でのデータ転送のためのデータ転送制御装置であって、各ノードの基本情報が格納される自己識別パケットが各ノードから転送されてきた場合に、該自己識別パケットを診断し、所与の管理機能を有するマネージャになるべきノードを診断結果に基づき自動検出する回路と、検出されたノードの識別番号を該ノードの自己識別パケットの基本情報の中から取得し、該識別番号を保持する第1の保持手段とを含むことを特徴とする。

【0009】本発明によれば、所与のマネージャになるべきノードの識別番号がハードウェアにより自動検出され、その識別番号が、例えばレジスタ、半導体メモリなどにより構成される第1の保持手段に保持される。従って、データ転送を制御するファームウェアなどの処理負担を軽減することが可能になる。これにより、実転送速度の向上や、装置の低コスト化などを期待できるようになる。

【0010】また本発明は、ノードの接続トポロジにおいてルートノードから遠いノードから順に自己識別パケットが転送されてくる場合において、自己識別パケットを転送してきたノードが、前記所与の管理機能を有するマネージャになれる資格を有するノードであると自己識別パケットの基本情報に基づき判断された場合には、該ノードの識別番号を前記第1の保持手段に上書きすることを特徴とする。このように自己識別パケットが送られてくる順序をうまく利用することで、マネージャになるノードを、より簡易に検出できるようになる。

【0011】また本発明は、下層から送られてくるステ

ータス情報に基づいて自己識別期間中か否かを検出する回路を含み、自己識別期間中に送られてきたパケットを自己識別パケットと見なし、該自己識別パケットに基づいてマネージャになるべきノードの自動検出を行うことを特徴とする。このようにすれば、処理対象となるパケットが自己識別パケットか否かを簡易に検出できるようになり、ハードウェアの小規模化を図れる。

【0012】また本発明は、マネージャとなるノードの識別番号が自身のノードの識別番号と一致した場合に、自身のノードが前記所与の管理機能を有するマネージャであることを示す情報を保持する第2の保持手段を含むことを特徴とする。このようにすれば、自身のノードが所与の管理機能を有するマネージャか否かを容易に知ることができるようになる。

【0013】また本発明は、自身のノードの識別番号が確定したことを条件にアクティブになる第1の信号と自己識別期間が終了したことを条件にアクティブになる第2の信号の両方を用いて、マネージャとなるノードの識別番号が自身のノードの識別番号と一致したか否かを判断することを特徴とする。第1の信号を用いることで、自己識別期間が終了した後に自身のノードの識別番号が確定した場合にも、適正に対処できるようになる。また、第2の信号を用いることで、自身のノードの識別番号が確定した後に、マネージャになるノードが変更した場合にも、適正に対処できるようになる。

【0014】また本発明は、サブアクションギャップを検出したことを条件にアクティブになる第3の信号を用いて、マネージャとなるノードの識別番号が自身のノードの識別番号と一致したか否かを判断することを特徴とする。このようにすれば、自身のノードの識別番号が必ず自己識別期間中に確定するような物理層のデバイスとの組み合わせにおいて、簡素な回路構成で、マネージャとなるノードの識別番号と自身のノードの識別番号の一致を適正に検出できるようになる。

【0015】また本発明は、前記第1の保持手段から前記所与の管理機能を有するマネージャになるノードの識別番号を読み出し、読み出された識別番号を利用してデータ転送を制御する手段を含むことを特徴とする。このようなデータ転送制御手段としては、例えば中央処理ユニット上で動くファームウェアなどを考えることができる。

【0016】なお本発明では、前記所与の管理機能を有するマネージャが、IEEE1394規格におけるアイソクロナスリソースマネージャであることが望ましい。

【0017】また本発明に係る電子機器は、上記のいずれかのデータ転送制御装置と、前記データ転送制御装置及びバスを介して他のノードから受信したデータに所与の処理を施す装置と、処理が施されたデータを出力又は記憶するための装置とを含むことを特徴とする。また本発明に係る電子機器は、上記のいずれかのデータ転送制

御装置と、前記データ転送制御装置及びバスを介して他のノードに送信するデータに所与の処理を施す装置と、処理が施されるデータを取り込むための装置とを含むことを特徴とする。

【0018】本発明によれば、他のノードから転送されたデータを電子機器において出力したり記憶したりする処理、電子機器において取り込んだデータを他のノードに転送したりする処理を高速化することが可能になる。また、本発明によれば、データ転送を制御するファームウェアなどの処理負担が軽減されるため、電子機器の低コスト化、小規模化などを図ることも可能になる。

【0019】

【発明の実施の形態】以下、本発明の好適な実施形態について図面を用いて詳細に説明する。

【0020】1. IEEE1394

まず、IEEE1394の概要について簡単に説明する。

【0021】1. 1 データ転送速度、接続トポロジー IEEE1394 (IEEE1394-1995、IEEE1394. A) では100~400Mbpsの高速なデータ転送が可能となっている (IEEE1394. Bでは800~3200Mbps)。また、転送速度が異なるノードをバスに接続することも許される。

【0022】各ノードはツリー状に接続されており、1つのバスに最大で63個のノードが接続可能になっている。なお、バスブリッジを利用すれば約64000個のノードを接続することも可能である。

【0023】電源が投入されたり、途中でデバイスの抜き差しが発生すると、バスリセットが発生し、接続トポロジーに関する情報が全てクリアされる。そして、バスリセット後、ツリー識別 (ルートノードの決定)、自己識別が行われる。その後、アイソクロナスリソースマネージャ、サイクルマスタ、バスマネージャ等の管理ノードが決定される。そして、通常の packets 転送が開始される。

【0024】1. 2 転送方式

IEEE1394では、packetsの転送方式として、信頼性が要求されるデータの転送に好適な非同期転送と、リアルタイム性が要求される動画や音声などのデータの転送に好適なアイソクロナス転送が用意されている。

【0025】図1 (A) に、非同期サブアクションの例を示す。1つのサブアクションは、調停、packets 転送、acknowledgment からなる。即ち、データ転送に先立って、まず、バスの使用権に関する調停が行われる。そしてソース (転送元) ノードからデスティネーション (転送先) ノードにpacketsが転送される。このpacketsのヘッダにはソースIDとデスティネーションIDが含まれる。デスティネーションノードは、このデスティネーションIDを読んで、自ノード宛のpacketsか否かを判断する。デスティネーションノードは、packets

トを受け取ると、ソースノードにacknowledgment (ACK) のpacketsを返す。

【0026】packets転送とACKの間にはacknowledgment ジャンプが存在する。また、1つのサブアクションと次のサブアクションの間にはサブアクションジャンプが存在する。そして、サブアクションジャンプに相当する一定のバス・アイドル時間が経過しないと、次のサブアクションの調停を開始できない。これによりサブアクション相互の衝突が回避される。

【0027】図1 (B) に、アイソクロナスサブアクションの例を示す。アイソクロナス転送はブロードキャスト (バスに接続される全てのノードに転送) で実行されるため、packets受信時にACKは返送されない。また、アイソクロナス転送では、ノードIDではなくチャンネル番号を使用してpackets転送が行われる。なお、サブアクション間にはアイソクロナスジャンプが存在する。

【0028】図1 (C) に、データ転送時のバスの様子を示す。アイソクロナス転送は、サイクルマスタが一定周期毎にサイクルスタートpacketsを発生することで開始する。これにより、1つのチャンネル当たり、125μs毎に少なくとも1つのpacketsを転送できるようになる。この結果、動画や音声などのリアルタイム性が要求されるデータの転送が可能になる。

【0029】非同期転送はアイソクロナス転送の合間に行われる。即ち、アイソクロナス転送の方が非同期転送よりも優先順位が高くなっている。これは、図1 (C) に示すように、アイソクロナスジャンプの時間を、非同期転送のサブアクションジャンプの時間よりも短くすることで実現される。

【0030】1. 3 ツリー識別

ツリー識別はバスリセットの後に行われる。ツリー識別によりノード間の親子関係やルートノードが決定される。

【0031】まず、リーフノード (1つのノードにしか接続されていないノード) が、隣接するノードに、ペアレントノーティファイを送る。例えば図2 (A) のようにノードA、B、C、D、Eが接続されている場合には、ノードAからノードBに、ノードD及びEからノードCに、ペアレントノーティファイ (PN) が送られる。

【0032】ペアレントノーティファイを受け取ったノードは、送り元のノードを自身の子と認知する。そして、チャイルドノーティファイをそのノードに送る。例えば図2 (A) では、ノードBからノードAに、ノードCからノードD及びEにチャイルドノーティファイ (CN) が送られる。これによってノードB、A間、ノードC、D間、ノードC、E間の親子関係が決定する。

【0033】ノードB、Cの親子関係は、どちらが先にペアレントノーティファイを送ったかで決定される。例

例えば図 2 (B) のように、ノード C の方が先にペアレントノティファイを送ると、ノード B が親になりノード C が子になる。

【0034】ポートの接続先の全てのノードが自身の子となるノードがルートになる。図 2 (B) では、ノード B がルートになる。なお、IEEE 1394 では、全てのノードがルートになる可能性がある。

【0035】1. 4 自己識別

ツリー識別の後、自己識別が行われる。自己識別においては、接続トポロジーにおいてルートノードから遠いノードから順にセルフ ID パケットが転送される。

【0036】より具体的には、例えば図 3 において、まず、ルートノード B のポート 1 (番号の小さいポート) に接続されるノード A が、セルフ ID パケット (自己識別パケット) を全てのノードにブロードキャストする。

【0037】次に、ルートノード B のポート 2 (番号が大きいポート) に接続されるノード C が選択され、このノード C のポート 1 (番号の小さいポート) に接続されるノード D がセルフ ID パケットをブロードキャストする。次に、ノード C のポート 2 (番号の大きいポート) に接続されるノード E がセルフ ID パケットをブロードキャストし、その後、ノード C がブロードキャストする。最後に、ルートであるノード B がセルフ ID パケットをブロードキャストし、自己識別が完了する。

【0038】セルフ ID パケットには各ノードの ID が含まれる。ブロードキャストを行う時点で他のノードから受け取ったセルフ ID パケットの個数が、この各ノードの ID となる。例えば図 3 では、ノード A がブロードキャストを行う時点では、どのノードもセルフ ID パケットを発していないため、ノード A の ID は 0 になる。ノード A は、この ID=0 をセルフ ID パケットに含ませてブロードキャストする。また、ノード D がブロードキャストを行う時点では、ノード A のみがセルフ ID パケットを発している。このため、ノード D の ID は 1 になる。同様に、ノード E、C、B の ID は、各々、2、3、4 になる。

【0039】図 4 (A) にセルフ ID パケットのフォーマットを示す。同図に示すようにセルフ ID パケットには各ノードの基本情報が含まれる。具体的には、各ノードの ID (PHY_ID)、リンク層がアクティブか否か (L)、ギャップカウント (gap_cnt)、転送速度 (sp)、アイソクロナスリソースマネージャになれる能力を有するか否か (C)、電力状態 (pwr)、ポートの状態 (p0、p1、p2) などに関する情報が含まれる。

【0040】なお、図 4 (B) に、ノードのポート数が 4 個以上の場合に使用されるセルフ ID パケット #1、#2、#3 のフォーマットを示す。ポート数が 4~11 個の場合にはセルフ ID パケット #0 (図 4 (A)) 及び #1 が、12~19 個の場合にはセルフ ID パケット

#0、#1 及び #2 が、20~27 個の場合にはセルフ ID パケット #0、#1、#2 及び #3 が使用されることになる。

【0041】また、図 4 (C)、(D) に、セルフ ID パケットと同様に、物理層のパケットであるリンクオンパケット、PHY 構成パケットのフォーマットを示す。

【0042】1. 5 アイソクロナスリソースマネージャ

アイソクロナスリソースマネージャ (IRM) は以下の管理機能を有する。

【0043】第 1 に、アイソクロナス転送に必要な種々のリソースを提供する。例えば、チャンネル番号レジスタや帯域幅レジスタを提供する。第 2 に、バスマネージャの ID を示すレジスタを提供する。第 3 に、バスマネージャがない場合に、簡易的なバスマネージャとなる役割を有する。

【0044】IRM になれる能力を有し (アイソクロナスリソースを管理する能力を有し)、且つ、動作状態になっている (リンク層がアクティブになっている) ノードの中で (IRM になれる資格を有するノードの中で)、ルートに最も近い (ID が最も大きい) ノードが IRM になる。より、具体的には、図 4 (A) のセルフ ID パケットにおいて、IRM になれる能力を有するか否かを示す C (CONTENDER) ビットと、リンク層がアクティブか否かを示す L (LINK_ACTIVE) ビットが共に 1 になっているノードの中で、ルートに一番近いノード (PHY_ID が一番大きいノード) が IRM になる。例えば、ルートノードのセルフ ID パケットの C ビットと L ビットが 1 の場合には、ルートノードが IRM になる。

【0045】1. 6 サイクルマスタ、バスマネージャ
サイクルマスタは、図 1 (C) に示すサイクルスタートパケットを送信する役割を有し、ルートノードがサイクルマスタになる。

【0046】バスマネージャは、トポロジーマップ (各ノードの接続状態) の作成、スピードマップの作成、バスの電力管理、サイクルマスタの決定、ギャップカウンットの最適化などの仕事を行う。

【0047】1. 7 プロトコル構成

図 5 を用いて、IEEE 1394 のプロトコル構成 (層構造) について説明する。

【0048】IEEE 1394 のプロトコルは、物理層、リンク層、トランザクション層により構成される。また、シリアルバスマネージメントは、物理層、リンク層、トランザクション層をモニターしたり制御したりするものであり、ノードの制御やバスのリソース管理のための種々の機能を提供する。

【0049】物理層は、リンク層により使用されるロジカルシンボルを電気信号に変換したり、バスの調停を行ったり、バスの物理的インターフェースを定義する。

【0050】リンク層は、アドレッシング、データチェック、データフレーミング、サイクル制御などを提供する。

【0051】トランザクション層は、リード、ライト、ロックなどのトランザクションを行うためのプロトコルを定義する。

【0052】物理層及びリンク層は、通常、データ転送制御装置（インターフェースチップ）などのハードウェアにより実現される。また、トランザクション層は、CPU上で動作するファームウェアや、ハードウェアにより実現される。

【0053】2. 全体構成

次に、本実施形態の全体構成について図6を用いて説明する。

【0054】図6において、PHYインターフェース10は、物理層のプロトコルを実現するPHYチップとのインターフェースを行う回路である。

【0055】リンクコア20は、リンク層のプロトコルやトランザクション層のプロトコルの一部を実現する回路である。レジスタ22は、これらのプロトコルを実現したりリンクコア20を制御するためのレジスタである。

【0056】FIFO(ATF)30、FIFO(ITF)32、FIFO(RF)34は、各々、非同期送信用、アイソクロナス送信用、受信用のFIFOであり、例えばレジスタや半導体メモリなどのハードウェアにより構成される。本実施形態では、これらのFIFO30、32、34の段数は非常に少ない。例えば1つのFIFOの段数は、好ましくは3段以下であり、更に好ましくは2段以下となる。

【0057】DMAC40、42、44は、各々、ATF、ITF、RF用のDMAコントローラである。これらのDMAC40、42、44を用いることで、CPU66に介入されることなく、RAM80とリンクコア20との間でのデータ転送が可能になる。なお、レジスタ46は、DMAC40、42、44などを制御するレジスタである。

【0058】ポートインターフェース50は、アプリケーション層のデバイス（例えばプリンタの印字処理を行うデバイス）とのインターフェースを行う回路である。本実施形態では、このポートインターフェース50を用いて、例えば8ビットのデータ転送が可能になっている。

【0059】FIFO(PF)52は、アプリケーション層のデバイスとの間でのデータ転送のためのFIFOであり、DMAC54は、PF用のDMAコントローラである。レジスタ56は、ポートインターフェース50やDMAC54を制御するレジスタである。

【0060】CPUインターフェース60は、データ転送制御装置をコントロールするCPU66とのインター

フェースを行う回路である。CPUインターフェース60は、アドレスデコーダ62、データ同期化回路63、割り込みコントローラ64を含む。クロック制御回路68は、本実施形態で使用されるクロックを制御するものであり、PHYチップから送られてくるSCLKや、マスタクロックであるHCLKが入力される。

【0061】バッファマネージャ70は、RAM80とのインターフェースを管理する回路である。バッファマネージャ70は、バッファマネージャの制御のためのレジスタ72、RAM80へのバス接続を調停する調停回路74、各種の制御信号を生成するシーケンサ76を含む。

【0062】RAM80は、ランダムアクセス可能なパケット記憶手段として機能するものであり、その機能は例えばSRAM、DRAMなどにより実現される。そして、本実施形態では、図7に示すように、このRAM80がヘッダ領域（広義には制御情報領域）とデータ領域に分離されている。そして、パケットのヘッダ（広義には制御情報）は図7のヘッダ領域に格納され、パケットのデータはデータ領域に格納される。

【0063】なおRAM80は、本実施形態のデータ転送制御装置に内蔵させることが特に望ましいが、その一部又は全部を外付けにすることも可能である。

【0064】バス90（或いはバス92、94）は、アプリケーションに接続されるものである（第1のバス）。またバス96（或いはバス98）はデータ転送制御装置をコントロールするためのものであり、データ転送制御装置をコントロールするデバイス（例えばCPU）に電氣的に接続される（第2のバス）。またバス100（或いはバス102、104、105、106、107、108、109）は、物理層のデバイス（例えばPHYチップ）に電氣的に接続されるものである（第3のバス）。また、バス110は、ランダムアクセス可能な記憶手段であるRAMに電氣的に接続されるものである（第4のバス）。

【0065】バッファマネージャ70の調停回路74は、DMAC40、DMAC42、DMAC44、CPUインターフェース60、DMAC54からのバスアクセス要求の調停を行う。そして、この調停結果に基づいて、各々、バス105、107、109、98、94のいずれかと、RAM80のバス110との間にデータの経路が確立される（第1、第2、第3のバスのいずれかと第4のバスとの間にデータ経路が確立される）。

【0066】本実施形態の1つの特徴は、ランダムアクセスが可能でありパケットを格納するRAM80を設けると共に、互いに分離されるバス90、96、100と、これらのバスをRAM80のバス110に接続するための調停回路74とを設けた点にある。

【0067】例えば図8に、本実施形態と構成の異なるデータ転送制御装置の例を示す。このデータ転送制御装

置では、リンクコア902は、PHYインターフェース900、バス922を介してPHYチップと接続される。また、リンクコア902は、FIFO904、906、908、CPUインターフェース910、バス920を介してCPU912に接続される。そして、CPU912は、バス924を介して、CPUにローカルなメモリであるRAM914に接続される。

【0068】なお、FIFO904、906、908は、図6のFIFO30、32、34と異なり、非常に段数の多いものとなる（例えば1つのFIFOが16段程度）。

【0069】図8の構成のデータ転送制御装置を用いた場合のデータ転送の手法について図9を用いて説明する。PHYチップ930を介して他のノードから送られてきた受信パケットは、バス922、データ転送制御装置932、バス920を介してCPU912が受け取る。そして、CPU912は、受け取った受信パケットをバス924を介してRAM914に書き込む。そして、CPU912は、受信パケットをアプリケーション層が使用できるように加工し、バス926を介してアプ

リケーション層のデバイス934に転送する。

【0070】一方、アプリケーション層のデバイス934からのデータを転送する場合には、CPU912は、このデータをRAM914に書き込む。そして、RAM914のデータにヘッダを付加することでIEEE1394に準拠したパケットを生成する。そして生成されたパケットは、データ転送制御装置932、PHYチップ930などを介して他のノードに送信される。

【0071】しかしながら、このようなデータ転送手法によると、CPU912の処理負荷が非常に重くなる。従って、ノード間を接続するシリアルバスの転送速度が高速になっても、CPU912の処理のオーバーヘッドなどに起因して、システム全体の実転送速度は低くなり、結局、高速なデータ転送を実現できない。

【0072】このような問題を解決する1つの手法として、図10に示すように、データ転送制御装置932とRAM914との間でのデータ転送や、RAM914とアプリケーション層のデバイス934との間でのデータ転送を、ハードウェアDMAにより実現する手法も考えられる。

【0073】しかしながら、この手法では、CPUバス928が、データ転送制御装置932、RAM914間でのデータ転送、RAM914、CPU912間でのデータ転送、RAM914、アプリケーション層デバイス934間でのデータ転送に使用されることになる。従って、システム全体のデータ転送の高速化を図ろうとすると、CPUバス928としてPCIバスのような高速なバスを使用しなければならなくなり、これは、データ転送制御装置を使用する電子機器の高コスト化を招く。

【0074】これに対して、本実施形態では図11に示

すように、データ転送制御装置120、アプリケーション層デバイス124間のバス90と、CPUバス96と、データ転送制御装置120、RAM80間のバス110とが互いに分離されている。従って、CPUバス96をデータ転送の制御のみに使用できるようになる。また、バス90を占有して、データ転送制御装置120、アプリケーション層デバイス124間でデータ転送を行うことができるようになる。例えば、データ転送制御装置120が組み込まれる電子機器がプリンタである場合には、バス90を占有して印字データを転送できるようになる。この結果、CPU66の処理負荷を軽減でき、システム全体の実転送速度を高めることができる。またCPU66として安価なものを採用できると共に、CPUバス96として高速なバスを使用する必要性がなくなる。このため、電子機器の低コスト化、小規模化を図れるようになる。

【0075】3. I RM自動検出

さて、本実施形態のデータ転送制御装置は、IRMの自動検出機能を有している。この機能は図6のリンクコア20により実現される。

【0076】図12に、リンクコア20の構成の一例を示す。なお図12では、IRM自動検出に関係ない回路ブロックについては省略している。

【0077】リンクコア20は、バス監視回路130、直列・並列変換回路132、IRM検出回路140を含む。そして、IRM検出回路140は、パケット診断回路142、PHYIDレジスタ144、IRMIDレジスタ146、WIRM生成回路148、WIRMレジスタ150を含む。

【0078】ここで、バス監視回路130は、PHYインターフェース10を介してPHYチップに接続される8ビット幅のデータバスD、2ビット幅のコントロールバスCTLを監視する回路である。

【0079】直列・並列変換回路132は、データバスDのデータを32ビットのデータに変換する回路である。例えば、転送速度が400Mbpsの場合には8ビットのデータが32ビットのデータに、200Mbpsの場合には4ビットのデータが32ビットのデータに、100Mbpsの場合には2ビットのデータが32ビットのデータに変換される。

【0080】パケット診断回路142は、セルフIDパケットなどのパケットを診断する回路である。

【0081】PHYIDレジスタ144は、自身のノードのID(PHY_ID)を保持するものである。また、IRMIDレジスタ146は、IRMになれる資格のあるノードのID(IRM_ID)が順次上書きされるレジスタである。なお、最終的には、IRMになったノードのIDがIRMIDレジスタ146に保持されることになる。

【0082】WIRM生成回路148は、PHY_ID

とIRM_IDが一致したか否かを検出し、一致した場合には、自身がIRMになったことを示す情報であるWIRM (WonIRM) を生成する。そして、このWIRMは、WIRMレジスタ150に保持される。

【0083】次に、IRMの自動検出処理の詳細について、図13のタイミング波形図を用いて説明する。図13は、ノード数が3であり、1番目にセルフIDパケットを転送したノードが自身のノードであり、3番目にセルフIDパケットを転送したノードがIRMになる場合の例である。

【0084】まず、バス監視回路130が、セルフID期間 (バスリセットから1回目のサブアクションギャップまでの期間) 中か否かを判断し、セルフID期間中の場合には図13のA1に示すように、バスリセット中であることを示す信号BRIPをHレベルにする。

【0085】セルフID期間中か否かは以下のようにして検出する。まず、バス監視回路130が、PHYチップからの2ビットのCTLを監視する。そして図14

(A) に示すようにCTLが(01)の場合には、図14(B) に示すようにPHYチップからデータバスDを介してステータス情報が送られてくると判断する。このPHYチップからDを介して送られてくるステータス情報の中のステータスビットの内容を調べることで、図14(C) に示すようにバスリセットがなされたか否か(ビット2)、サブアクションギャップが来たか否か(ビット1)などを判断できる。そして、バス監視回路130は、バスリセットがなされたと判断すると、図13に示すようにBRIPをHレベルにし、サブアクションギャップが来たと判断すると、BRIPをLレベルに戻す。

【0086】このBRIPのレベルを監視することで、パケット診断回路142は、セルフID期間中か否かを知ることができるようになる。そして、このセルフID期間に送られてきたパケットをセルフIDパケットと見なし、これらのセルフIDパケットを診断することでIRMの自動検出を行うことができるようになる。

【0087】また、バス監視回路130は、CTL及びDを用いて自身のノードのIDも取得する。即ち、PHYチップは、セルフID期間において自身のIDが確定すると、このIDを含むセルフIDパケットをブロードキャストすると共に、PHYチップが内蔵するレジスタにこのIDを格納する。このPHYチップのレジスタの内容は、図14(C) に示すようにステータス情報のステータスビット8~15を見ることで知ることができる。従って、バス監視回路130は、自身のIDが確定すると、CTL及びDを調べることで、そのIDを取得できる。そして、取得したIDは、SD (Status Data) としてPHYIDレジスタ144に出力される。なおSSは、SDの取り込みタイミングを知らせるためのストロープ信号である。

【0088】例えば図13では、1番目にセルフIDパケットを転送したノードが自身のノードである。従って、取得した自身のIDを知らせるSDがA2のタイミングで出力され、PHYIDレジスタ144は、このSDをSSにより取り込む。これにより、A3に示すように、PHYIDレジスタ144は、自身のノードのIDであるPHY_IDを保持できるようになる。

【0089】さて、図14(B) に示すように、CTLが(10)である場合には受信状態になり、PHYチップからデータバスDを介してセルフIDパケットが送られてくる。直列・並列変換回路132は、これらのセルフIDパケットのデータを32ビットのデータであるDIに変換し、パケット診断回路142、IRMIDレジスタ146に出力する。

【0090】なお、DIEは、DIのデータが有効か無効かを示す信号である (DIEがHレベルの時に有効)。このDIEを調べることでパケット診断回路142はパケットの区切りを知ることができる。また、DISは、DIの取り込みタイミングを知らせるためのストロープ信号である。

【0091】DIを受けたパケット診断回路142は、DIに含まれる各ノードのセルフIDパケットを診断する。即ち図4(A) のセルフIDパケットのCビットとLビットを調べる。そしてIRMになれる能力を有するか否かを示すCビットが1であり、且つ、リンク層がアクティブか否かを示すLビットが1である場合には、そのセルフIDパケットを送ったノードはIRMになれる資格があるとして、GID (GetID) をHレベルにする。

【0092】例えば図13では、1番目にセルフIDパケットを送ったノード (自身のノード) が、IRMになれる資格があるため、A4のタイミングでGIDがHレベルになる。そして、直列・並列変換回路132からDIを受けているIRMIDレジスタ146は、このGIDをストロープ信号として、A5に示すように、1番目のセルフIDパケットに含まれるIDを取得し、これを保持する。

【0093】また図13では、3番目にセルフIDパケットを送ったノードも、IRMになれる資格を有するため、A6のタイミングでGIDがHレベルになる。そして、IRMIDレジスタ146は、このGIDをストロープ信号として、A7に示すように、3番目のセルフIDパケットに含まれるIDを取得し、これを前の内容に上書きする。そして、図13の場合には総ノード数が3個であるため、3番目にセルフIDを送ったノードがIRMに確定する。

【0094】即ち、セルフIDパケットはルートから遠いノードから順 (IDが小さいノードから順) に送られてくる。また、IRMになるのは、IRMになれる資格 (Cビット及びLビットが共に1) を有するノードの中

でルートに最も近いノードである（ルートノードがIRMになれる資格を有する場合にはルートノードがIRM）。従って、IRMになれる資格を有するノードのIDをIRMIDレジスタ146に上書きしてゆけば、最終的には、IRMに確定するノードのIDをIRMIDレジスタ146に保持することが可能になる。

【0095】以上のようにして本実施形態では、IRMになるノードのIDをハードウェアにより自動検出し、保持することに成功している。

【0096】また、本実施形態では、IRMになるノードが自身のノードであるか否かを判定し、自身のノードである場合にはそのことを示す情報を保持している。

【0097】即ち、PHYIDレジスタ144は、自身のノードのIDを示すPHY_IDをWIRM生成回路148に出力し、IRMIDレジスタ146は、IRMになるノード（正確にはIRMになる可能性のあるノード）のIDを示すIRM_IDをWIRM生成回路148に出力する。

【0098】WIRM生成回路148は、これらのPHY_IDとIRM_IDを比較し、一致した場合にはWIRM（WonIRM）をHレベルにし、一致しない場合にはWIRMをLレベルにする。そして、このWIRMはWIRMレジスタ150に保持される。

【0099】この際、本実施形態では、自身のノードのIDが確定したことを条件にアクティブになる信号であるGPID（GotPhyID）と、セルフID期間が終了したことを条件にアクティブになる信号であるESID（EndSelfID）の両方を用いて、PHY_IDとIRM_IDを比較している。

【0100】ここで、前者のGPIDは、PHYIDレジスタ144が、例えば信号SSなどを用いて生成する。また後者のESIDは、パケット診断回路142が、例えば信号BRIPなどを用いて生成する。

【0101】例えば図13では、A8に示すように、GPIDがHレベルになったタイミングでPHY_IDとIRM_IDが比較される。そして比較の結果、両者が一致しているため、WIRMはHレベルになる。またA9に示すように、ESIDがHレベルになったタイミングでも、PHY_IDとIRM_IDが比較される。そして比較の結果、両者が不一致であるため、WIRMはLレベルに戻る。

【0102】即ち、IRMになるノードは、全てのノードのセルフIDパケットを調べないと確定しない。このため、GPIDがアクティブになるタイミングのみならず、ESIDがアクティブになるタイミング（セルフID期間が終了したタイミング）でも、PHY_IDとIRM_IDが一致するか否かを調べる必要がある。

【0103】図15に、3番目にセルフIDパケットを転送したノードが自身のノードであり、この自身のノードがIRMになる場合の例を示す。この場合、B1に示

すようにESIDがアクティブになったタイミング、即ちセルフID期間が終了したタイミングでは、自身のノードのIDは確定していないことも有り得る。従って、このタイミングでPHY_IDとIRM_IDを比較すると、これらは一致していないと判断され、WIRMはLレベルのままになる。

【0104】一方、B2に示すように、GPIDがアクティブになったタイミング、即ち自身のノードが確定したタイミングで、再度PHY_IDとIRM_IDを比較すると、これらは一致している。このため、B3に示すように、WIRMはHレベルになる。これにより、3番目にセルフIDパケットを送った自身のノードがIRMになったことを、適正に検出できるようになる。

【0105】なお、GPIDがアクティブになるタイミング、即ち自身のノードのIDが確定するタイミングが、セルフID期間の終了よりも以前になるようなPHYチップ（物理層のデバイス）も存在し得る。このようなPHYチップと本実施形態のデータ転送制御装置とを組み合わせる場合には、セルフID期間の終了、即ちサブアクションギャップの検出をもって、自身のノードがIRMになったか否かを判断できる。

【0106】以上のように本実施形態によれば、IRMになるノードのIDを自動検出し、そのIDをIRMIDレジスタ146に保持できると共に、自身のノードがIRMになったことを示す情報（WIRM）もWIRMレジスタ150に保持できる。これにより、CPU66上で動作するファームウェアなどの処理負担を格段に軽減することに成功している。

【0107】即ち、アイソクロナス転送を行う場合には、アイソクロナス転送を行うチャンネルの番号と帯域幅をIRMから取得する必要がある。従って、ファームウェア（トランザクション層）は、IRMのIDを知る必要がある。

【0108】しかしながら、これまでは、IRMのIDをハードウェアにより自動検出できなかったため、ファームウェアが、セルフIDパケットに基づいてトポロジーマップを作成し、このトポロジーマップに基づいてIRMのIDを検出する必要があった。

【0109】図8の構成のデータ転送制御装置を例にとれば、CPU912で動作するファームウェアは図16（A）に示すような処理を行う必要がある。即ち、ファームウェアは、セルフIDパケットを受信すると、これをCPU912にローカルに設けられたRAM914に格納する（ステップS1、S2）。そして、ファームウェアは、全てのセルフIDパケットの受信が完了すると（ステップS3）、RAM914から、受信したセルフIDパケットをリードする（ステップS4）。そして、トポロジーマップを作成するなどしてIRMのIDをソフトウェアにより検出する（ステップS5）。このようにIRMのIDを検出する処理は、ファームウェアにと

って非常に負荷の重い処理となる。

【0110】これに対して、本実施形態によれば、図16(B)のステップT1に示すように、ファームウェアは、IRMのIDを知るためには図12のIRMIDレジスタ146をリードするだけでよい。従って、ファームウェアの処理負荷が非常に軽くなり、システム全体の実転送速度の向上を期待できるようになる。

【0111】また、図16(A)では、受信した全てのセルフIDパケットをRAMに格納する必要がある。セルフIDパケットは、ノードの個数に応じた個数だけ必要であり、バスに63個のノードが接続されると例えば最小で63個のセルフIDパケットを保持する必要がある。従って、セルフIDパケットをそのままRAMに格納すると、RAMの空き容量が減ってしまう。

【0112】これに対して、本実施形態によれば、セルフIDパケットをRAMに格納することなく、IRMを自動検出することが可能になる。従って、RAMの使用容量を節約することが可能となる。

【0113】なお、本実施形態においても、他の用途に使用するためにセルフIDパケットをRAMに格納するようにしてもよい。この場合には、ヘッダ（広義には制御情報）と複数のセルフIDパケットの並びからなるデータとによりフレームが構成されるパケットを生成し、このパケットをRAMに格納するようにすることが望ましい。更に、この場合、図4(A)、(B)のセルフIDパケットの2クワドレット目のエラー検出情報（最初の32ビットの反転）の部分削除したものの並びによりデータを構成することが望ましい。このようにすることで、生成されるパケットのサイズを更に小さくすることが可能になる。

【0114】4. 電子機器

次に、本実施形態のデータ転送制御装置を含む電子機器の例について説明する。

【0115】例えば図17(A)に電子機器の1つであるプリンタの内部ブロック図を示し、図18(A)にその外観図を示す。CPU（マイクロコンピュータ）510はシステム全体の制御などを行う。操作部511はプリンタをユーザが操作するためのものである。ROM516には、制御プログラム、フォントなどが格納され、RAM518はCPU510のワーク領域として機能する。表示パネル519はプリンタの動作状態をユーザに知らせるためのものである。

【0116】PHYチップ502、データ転送制御装置500を介して、パーソナルコンピュータなどの他のノードから送られてきた印字データは、バス504を介して印字処理部512に直接送られる。そして、印字データは、印字処理部512にて所与の処理が施され、プリントヘッダなどからなる印字部（データを出力するための装置）514により紙に印字されて出力される。

【0117】図17(B)に電子機器の1つであるスキ

ャナの内部ブロック図を示し、図18(B)にその外観図を示す。CPU520はシステム全体の制御などを行う。操作部521はスキャナをユーザが操作するためのものである。ROM526には制御プログラムなどが格納され、RAM528はCPU520のワーク領域として機能する。

【0118】光源、光電変換器などからなる画像読み取り部（データを取り込むための装置）522により原稿の画像が読み取られ、読み取られた画像のデータは画像処理部524により処理される。そして、処理後の画像データがバス505を介してデータ転送制御装置500に直接送られる。データ転送制御装置500は、この画像データにヘッダなどを付加することでパケットを生成し、PHYチップ502を介してパーソナルコンピュータなどの他のノードに送信する。

【0119】図17(C)に電子機器の1つであるCD-Rドライブの内部ブロック図を示し、図18(C)にその外観図を示す。CPU530はシステム全体の制御などを行う。操作部531はCD-Rをユーザが操作するためのものである。ROM536には制御プログラムなどが格納され、RAM538はCPU530のワーク領域として機能する。

【0120】レーザ、モータ、光学系などからなる読み取り&書き込み部（データを取り込むための装置又はデータを記憶するための装置）533によりCD-R532から読み取られたデータは、信号処理部534に入力され、エラー訂正処理などの所与の信号処理が施される。そして、信号処理が施されたデータが、バス506を介してデータ転送制御装置500に直接送られる。データ転送制御装置500は、このデータにヘッダなどを付加することでパケットを生成し、PHYチップ502を介してパーソナルコンピュータなどの他のノードに送信する。

【0121】一方、PHYチップ502、データ転送制御装置500を介して、他のノードから送られてきたデータは、バス506を介して信号処理部534に直接送られる。そして、信号処理部534によりこのデータに所与の信号処理が施され、読み取り&書き込み部533によりCD-R532に記憶される。

【0122】なお、図17(A)、(B)、(C)において、CPU510、520、530の他に、データ転送制御装置500でのデータ転送制御のためのCPUを別に設けるようにしてもよい。

【0123】本実施形態のデータ転送制御装置を電子機器に用いることで、高速なデータ転送が可能になる。従って、ユーザがパーソナルコンピュータなどによりプリントアウトの指示を行った場合に、少ないタイムラグで印字が完了するようになる。また、スキャナへの画像取り込みの指示の後に、少ないタイムラグで読み取り画像をユーザは見るができるようになる。また、CD-

Rからのデータの読み取りや、CD-Rへのデータの書き込みを高速に行うことができるようになる。更に、例えば1つのホストシステムに複数の電子機器を接続して利用したり、複数のホストシステムに複数の電子機器を接続して利用したりすることも容易になる。

【0124】また本実施形態のデータ転送制御装置を電子機器に用いることで、CPU上で動作するファームウェアの処理負荷が軽減される。従って、安価なCPUや低速のバスを用いることが可能になり、電子機器の低コスト化、小規模化を図ることが可能になる。

【0125】なお本実施形態のデータ転送制御装置を適用できる電子機器としては、上記以外にも例えば、種々の光ディスクドライブ（CDROM、DVD）、光磁気ディスクドライブ（MO）、ハードディスクドライブ、TV、VTR、ビデオカメラ、オーディオ機器、電話機、プロジェクタ、パーソナルコンピュータ、電子手帳、ワードプロセッサなど種々のものを考えることができる。

【0126】なお、本発明は本実施形態に限定されず、本発明の要旨の範囲内で種々の変形実施が可能である。

【0127】例えば、本発明のデータ転送制御装置の構成は、図6に示す構成が特に望ましいが、これに限定されるものではない。例えば、図8に示すような構成を採用することも可能である。

【0128】また、本発明における所与の管理機能をするマネージャは、IEEE1394規格におけるアイソクロナスリソースマネージャであることが特に望ましいが、これに限定されるものではない。

【0129】また、本発明は、IEEE1394規格でのデータ転送に適用されることが特に望ましいが、これに限定されるものではない。例えばIEEE1394と同様の思想に基づく規格やIEEE1394を発展させた規格におけるデータ転送にも本発明は適用できる。

【0130】

【図面の簡単な説明】

【図1】図1（A）、（B）、（C）は、非同期転送とアイソクロナス転送について説明するための図である。

【図2】図2（A）、（B）は、ツリー識別について説明するための図である。

【図3】自己識別について説明するための図である。

【図4】図4（A）、（B）、（C）、（D）は、セルブIDパケットなどの物理層のパケットのフォーマットを示す図である。

【図5】IEEE1394のプロトコル構成について示す図である。

【図6】本実施形態のデータ転送制御装置の構成例を示す図である。

【図7】ヘッダ（制御情報）領域とデータ領域の分離について説明するための図である。

【図8】本実施形態の比較例の構成例について示す図で

ある。

【図9】図8の構成によるデータ転送の手法について説明するための図である。

【図10】データ転送の手法の他の例について説明するための図である。

【図11】本実施形態のデータ転送の手法について説明するための図である。

【図12】リンクコア（IRM検出回路）の構成例について示す図である。

10 【図13】IRM検出の詳細な処理例について説明するためのタイミング波形図である。

【図14】図14（A）、（B）、（C）は、コントロールバスCTL、データバスD、ステータス情報（ステータスビット）について説明するための図である。

【図15】IRM検出の詳細な処理例について説明するためのタイミング波形図である。

【図16】図16（A）、（B）は、比較例と本実施形態のファームウェアの処理について比較するための図である。

20 【図17】図17（A）、（B）、（C）は、種々の電子機器の内部ブロック図の例である。

【図18】図18（A）、（B）、（C）は、種々の電子機器の外観図の例である。

【符号の説明】

10 PHYインターフェース

20 リンクコア

22 レジスタ

30 FIFO（ATF）

32 FIFO（ITF）

30 34 FIFO（RF）

40 DMAC（ATF用）

42 DMAC（ITF用）

44 DMAC（RF用）

50 ポートインターフェース

52 FIFO（PF）

54 DMAC（PF用）

56 レジスタ

60 CPUインターフェース

62 アドレスデコーダ

40 63 データ同期化回路

64 割り込みコントローラ

66 CPU

68 クロック制御回路

70 バッファマネージャ

72 レジスタ

74 調停回路

76 シーケンサ

80 RAM（パケット記憶手段）

90、92、94 バス（第1のバス）

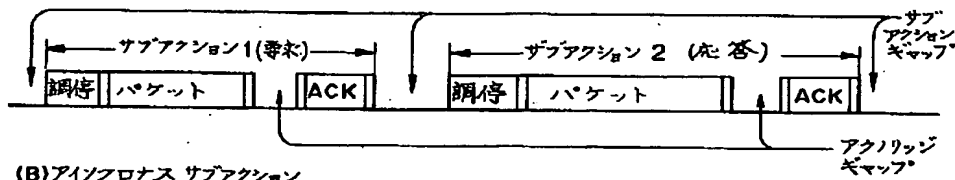
50 96、98 バス（第2のバス）

100、102、104、105、
 106、107、108、109 バス（第3のバス）
 110 バス（第4のバス）
 120 データ転送制御装置
 122 PHYチップ
 124 アプリケーション層のデバイス
 130 バス監視回路

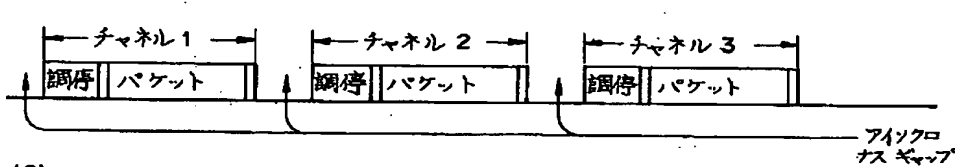
* 132 直列・並列変換回路
 140 I RM検出回路
 142 パケット診断回路
 144 PHYIDレジスタ
 146 I RMIDレジスタ（第1の保持手段）
 148 W I RM生成回路
 * 150 W I RMレジスタ（第2の保持手段）

【図1】

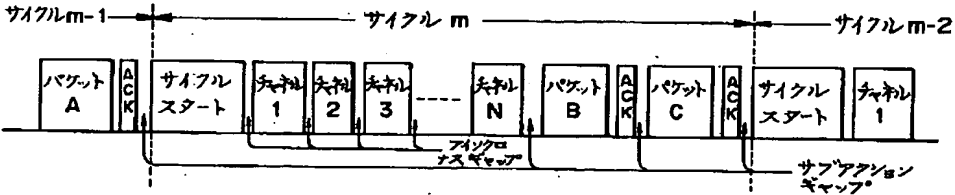
(A)非同期サブアクション



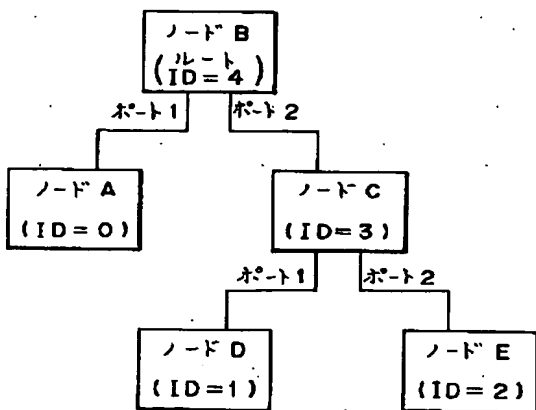
(B)アイソクロナスサブアクション



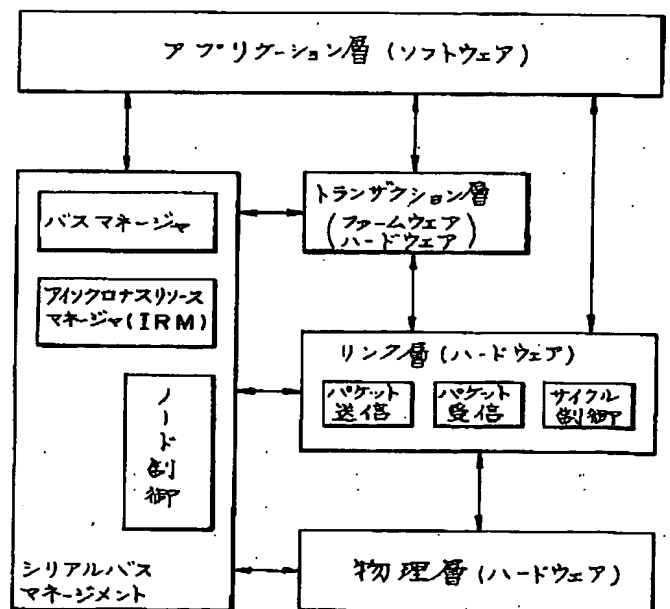
(C)



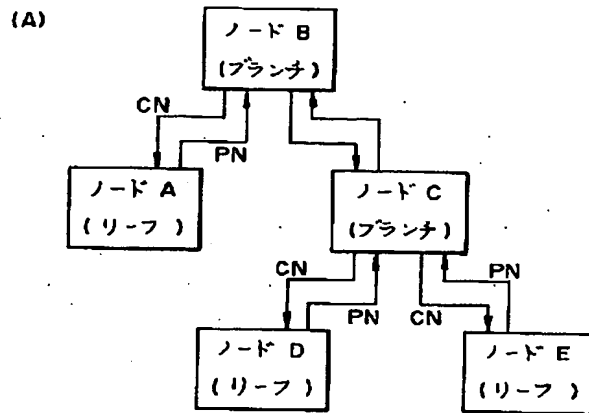
【図3】



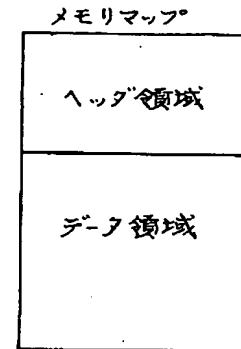
【図5】



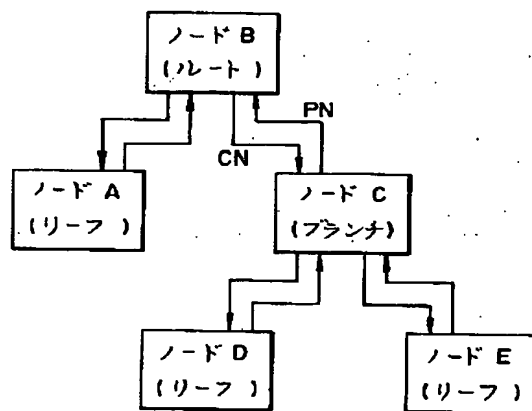
【図2】



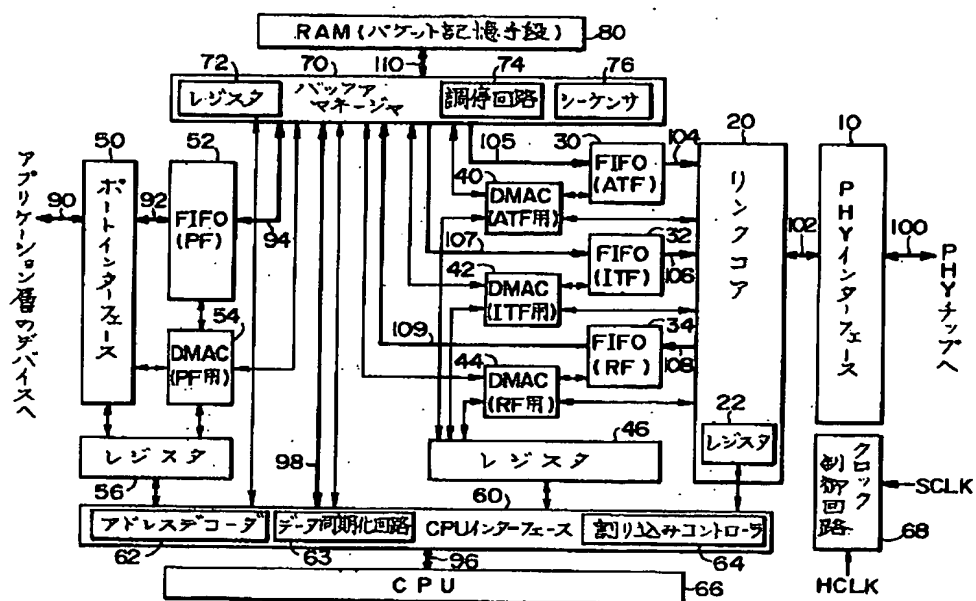
【図7】



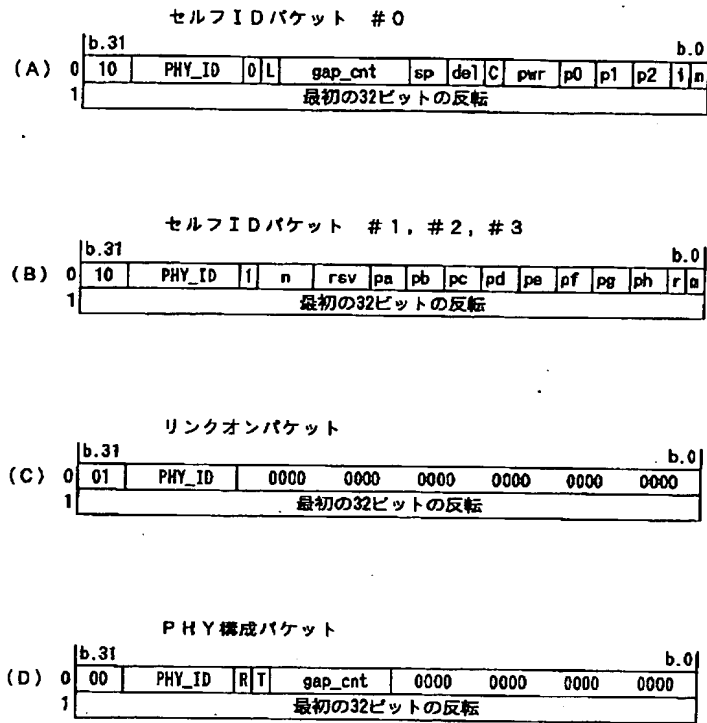
(B)



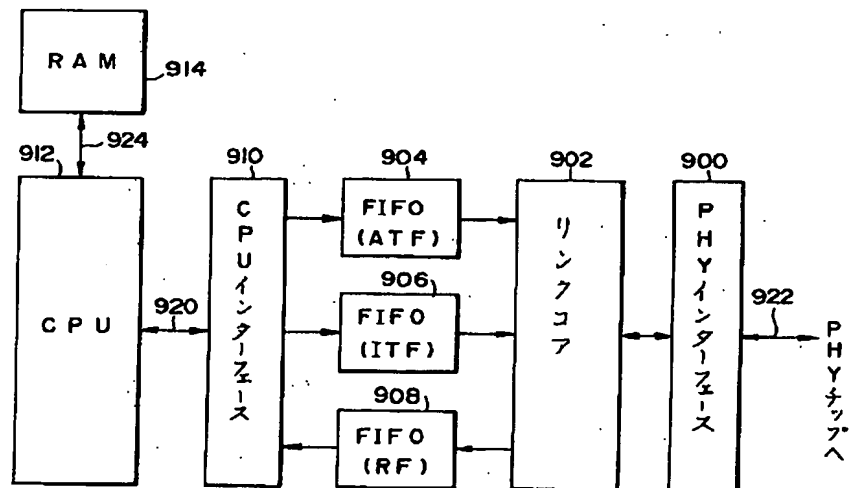
【図6】



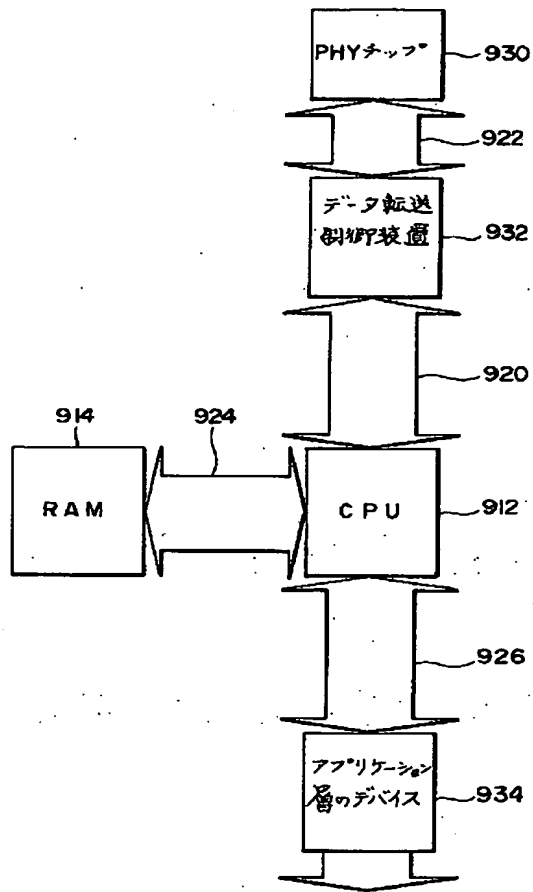
【図4】



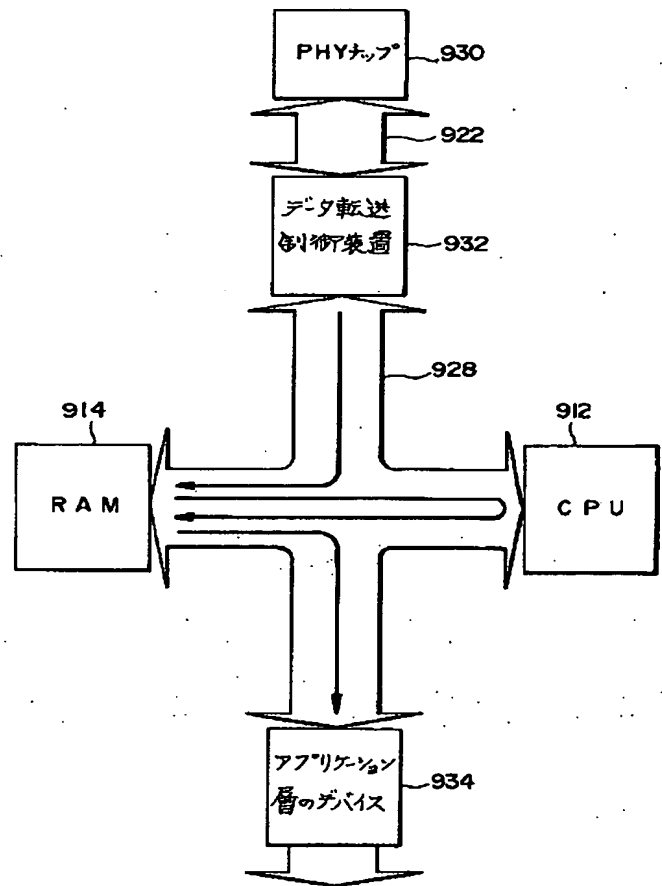
【図8】



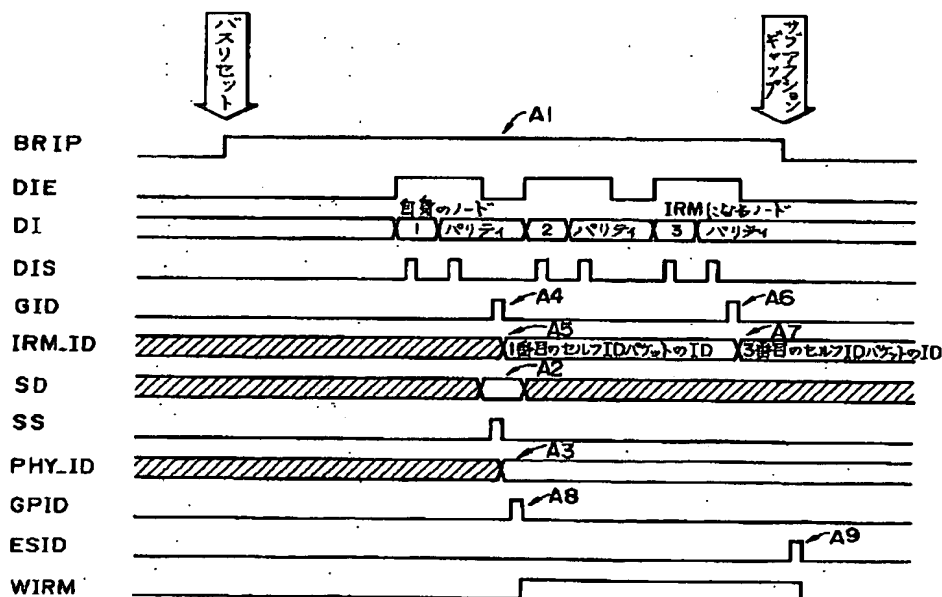
【図9】



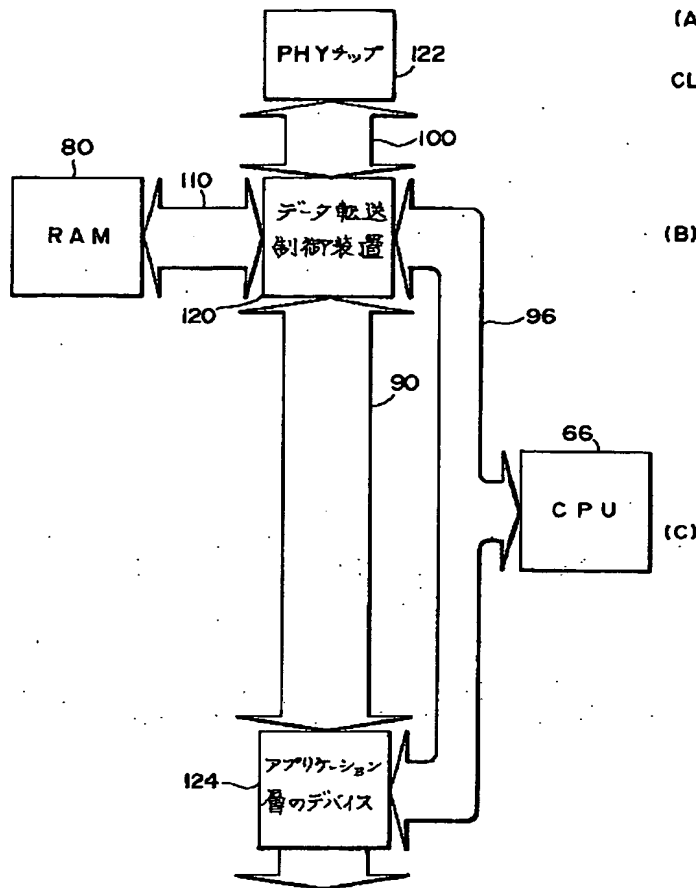
【図10】



【図13】

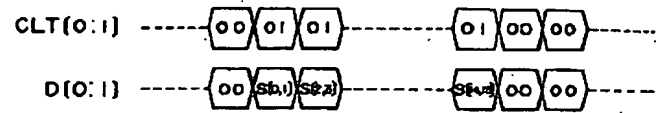


【図11】



【図14】

(A)



(B)

PHYチップ駆動時

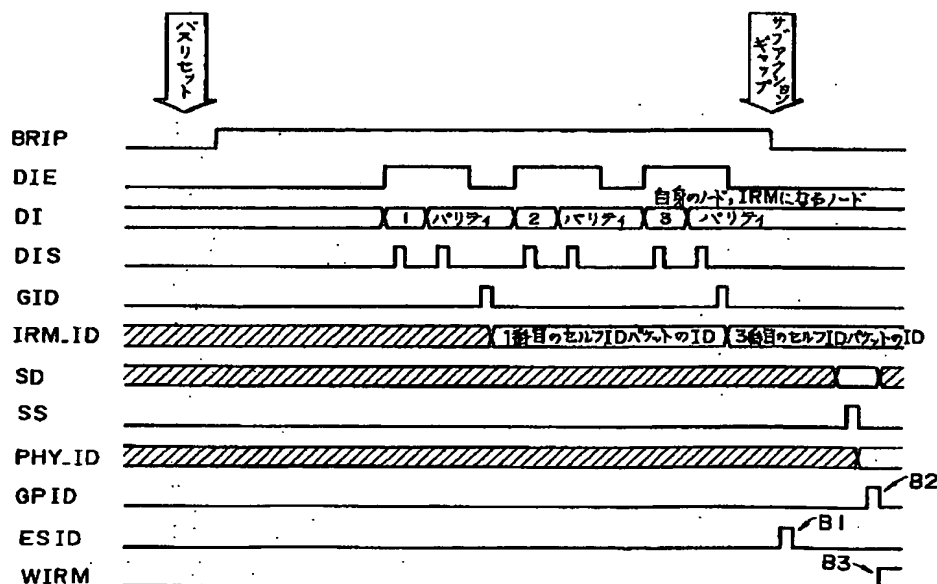
CTL	NAME
00	アイドル
01	ステータス
10	受信
11	転送

(C)

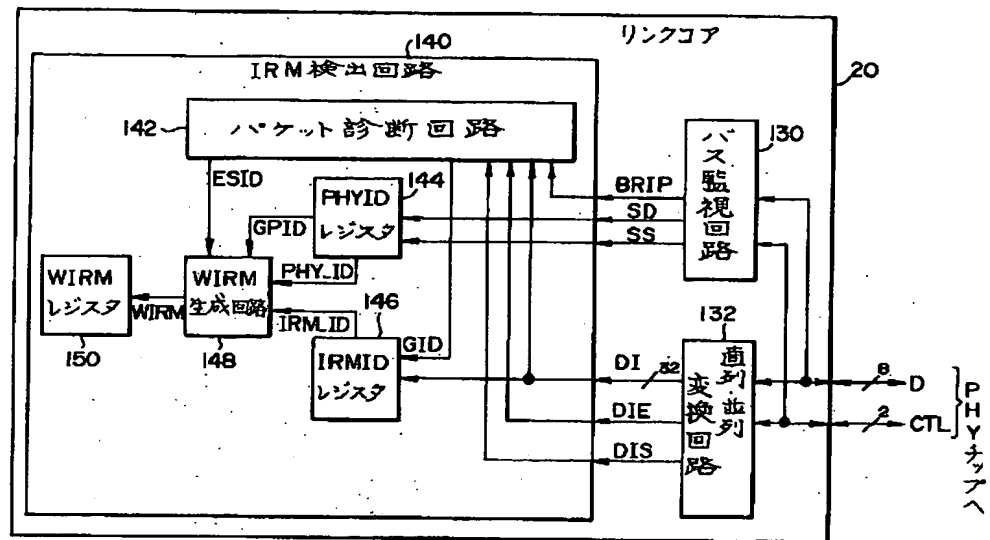
ステータスビット

BIT	NAME
0	調停リセットギャップ
1	サブアクションギャップ
2	バスリセット
3	PHY割り込み
4~7	アドレス
8~15	データ

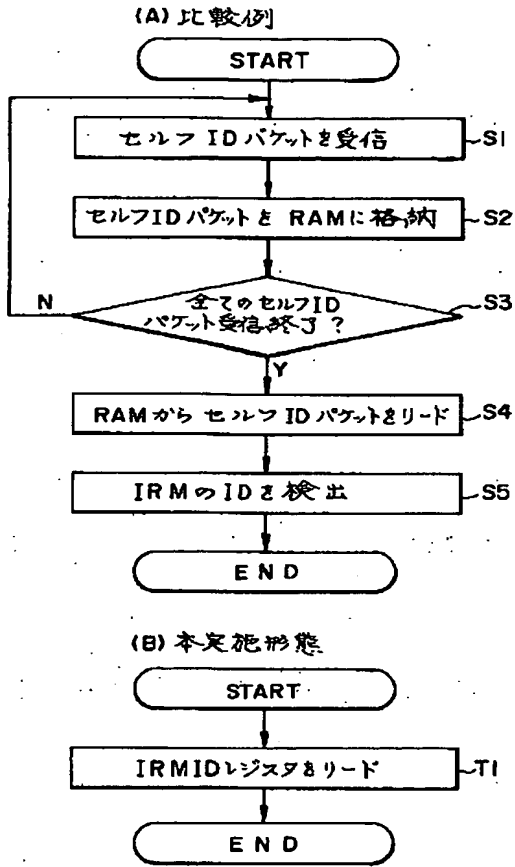
【図15】



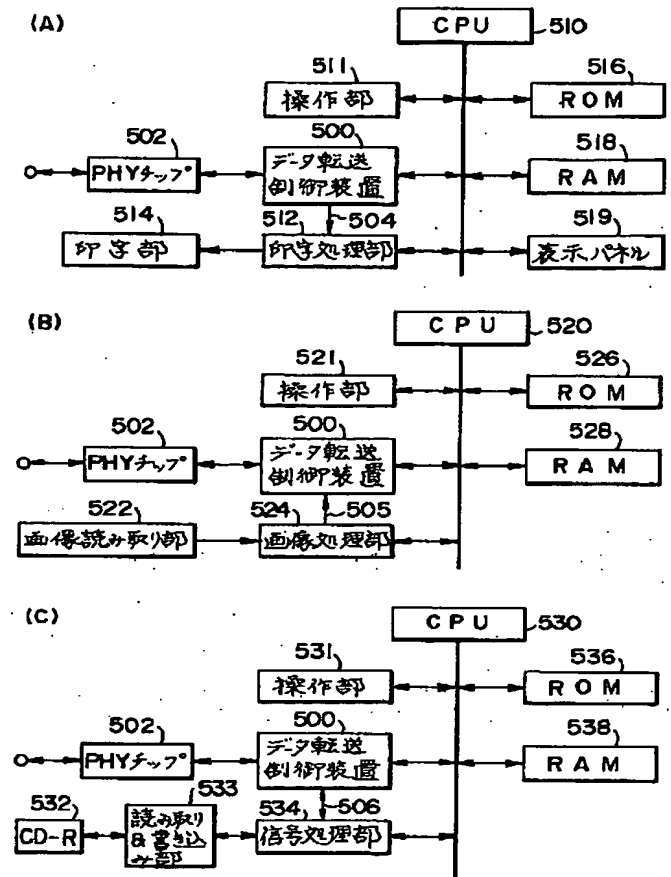
【図12】



【図16】



【図17】



【図 18】

